311-CD-100-001

EOSDIS Core System Project

Access Control List Database Design and Schema Specifications for the ECS Project

Draft

This document has not yet been approved by the Government for general use or distribution.

January 1998

Raytheon Systems Company
Upper Marlboro, Maryland

Access Control List
Database Design and Schema Specifications for the ECS Project

Draft

January 1998

Prepared Under Contract NAS5-60000
CDRL Item #050

RESPONSIBLE ENGINEER

Mary Armstrong for Maureen Muganda /s/ 12/31/97

Maureen Muganda Date
EOSDIS Core System Project

SUBMITTED BY

Terry Fisher /s/ 12/31/97

Terry Fisher, ECS CCB Chairman Date
EOSDIS Core System Project

Raytheon Systems Company
Upper Marlboro, Maryland

This page intentionally left blank.

Preface

This document describes the data design and database specification for the Access Control List (ACL) subsystem. It is one of ten documents comprising the detailed database design specifications for each of the ECS subsystems.

The subsystem database design specifications for the as delivered system include:

311-CD-100-001	Release 2.0 Access Control (ACL) Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-101-001	Release 2.0 Data Distribution (DDIST) Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-102-001	Release 2.0 Data Management (DM) Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-103-001	Release 2.0 Ingest Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-104-001	Release 2.0 Interoperability Subsystem (IOS) Database Design and Database Schema Specifications for the ECS Project
311-CD-105-001	Release 2.0 Management Support Subsystem (MSS) Database Design and Database Schema Specifications for the ECS Project
311-CD-106-001	Release 2.0 Planning and Data Processing Subsystem (PDPS) Database Design and Database Schema Specifications for the ECS Project
311-CD-107-001	Release 2.0 Science Data Server (SDSRV) Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-108-001	Release 2.0 Storage Management (STMGMT) Subsystem Database Design and Database Schema Specifications for the ECS Project
311-CD-109-001	Release 2.0 Subscription Server (SUBSRV) Subsystem Database Design and Database Schema Specifications for the ECS Project

This submittal meets the milestone specified in the Contract Data Requirements List (CDRL) of NASA Contract NAS5-60000. It is a formal contract deliverable with an approval code 1. It requires Government review and approval prior to acceptance and use. This document is under ECS contractor configuration control. Once approved, contractor approved changes will be handled in accordance with Class I and lass II change control requirements described in the EOS Configuration Management Plan, and changes to this document shall be made by Document Change Notice (DCN) or by complete revision.

Entity Relationship Diagrams (ERDs) presented in this document have been exported directly from tools and some cases contain too much detail to be easily readable within hard copy page constraints. The reader is encouraged to view these drawings on-line using the Portable Document Format (PDF) electronic copy available via the ECS Data Handling System (ECS) on the world-wide web at http://edhs1.gsfc.nasa.gov.

Any questions should be addressed to:

Data Management Office
The ECS Project Office
Raytheon Systems Company
1616 McCormick Drive
Upper Marlboro, MD 20774-5372

Abstract

This document outlines “as-built” database design and database schema of the Access Control List database including the physical layout of the database and initial installation parameters.

Keywords: data**,** database, design, configuration, database installation, scripts, security, data model, data dictionary, replication, performance tuning, SQL server, database security, replication, database scripts

This page intentionally left blank.

Change Information Page

List of Effective Pages	
Page Number	**Issue**
Title	Draft
iii through viii	Draft
1-1 and 1-2	Draft
2-1 and 2-2	Draft
3-1 and 3-2	Draft
4-1 through 4-12	Draft
5-1 and 5-2	Draft
6-1 and 6-2	Draft
7-1 and 7-2	Draft
8-1 and 8-2	Draft
AB-1 through AB-8	Draft

Document History			
Document Number	**Status/Issue**	**Publication Date**	**CCR Number**
311-CD-100-001	Draft	January 1998	97-1755

This page intentionally left blank.

Table of Contents

1. Introduction

2. Related Documents

3. Database Configurations

4. Data Design

5. Performance and Tuning Factors

6. Database Security

7. Replication

8. Scripts

List of Figures

Abbreviations and Acronyms

This page is intentionally left blank.

1. Introduction

1.1. Identification

This Access Control List (ACL) Database Design and Database Schema Specification document, Contract Data Requirement List (CDRL) Item Number 050, whose requirements are specified in Data Item description (DID_311/DV1, is a required deliverable under the Earth Observing System (EOS) Data and Information System (EOSDIS) Core System (ECS), Contract NAS5-60000.

1.2. Scope

The ACL Database Design and Database Schema Specification document describes the data design and database specifications to support the data requirements of Release 2 Drop 3 ACL software.

1.3. Purpose

The purpose of the ACL Database Design and Database Schema Specification document is to support the maintenance of ACL data and databases throughout the life cycle of ECS. This document communicates the database implementation in sufficient detail to support ongoing configuration management.

1.4. Audience

This document is intended to be used by ECS maintenance and operations staff. The document is organized as follows:

- Section 1 provides information regarding the identification, purpose, scope and audience of this document.
- Section 2 provides a listing of the related documents, which were used as a source of information for this document.
- Section 3 provides a mapping of data bases to hardware components.
- Section 4 contains the ACL physical data model which is the database tables, triggers, stored procedures, and flat files.
- Section 5. provides a description of database performance and tuning features such as indexes, caches, and data segments.
- Section 6 provides a description of the security infrastructure used and list of the users, groups, and permissions available upon initial installation.
- Section 7 contains replication design and implementation details.
- Section 8 provides a description of database and database related scripts used for installation, de-installation, backup/recovery, and other miscellaneous functions.

This page intentionally left blank.

2. Related Documents

2.1. Applicable Documents

The following documents, including Internet links, are referenced in the ACL Database Design and Database Schema Specification, or are directly applicable, or contain policies or other directive matters that are binding upon the content of this volume. Internet links cannot be guaranteed for accuracy or currency.

920-TDG-009 Release B0 GSFC DAAC Database Information

920-TDN-009 Release B0 NSIDC DAAC Database Information

920-TDE-009 Release B0 EDC DAAC Database Information

920-TDL-009 Release B0 LARC DAAC Database Information

920-TDS-009 Release B0 SMC DAAC Database Information

920-TDM-009 Release B0 Mini-DAAC Database Information

2.2. Information Documents

The following documents, although not directly applicable, amplify or clarify the information presented in this document. These documents are not binding on this document.

To Be Supplied (TBS)

This page intentionally left blank.

3. Database Configurations

3.1. Server Configurations

The database configuration of the ACL server varies from DAAC to DAAC based on individualized DAAC requirements and hardware availability. These DAAC-specific database configurations are detailed on the following documents:

920-TDG-00X Release B0 GSFC DAAC Database Information

920-TDN- 00X Release B0 NSIDC DAAC Database Information

920-TDE-00X Release B0 EDC DAAC Database Information

920-TDL-00X Release B0 LARC DAAC Database Information

920-TDS-00X Release B0 SMC DAAC Database Information

920-TDM-00X Release B0 Mini-DAAC Database Information

These documents are maintained as part of the ECS baseline and available on the world-wide web at the URL http://pete.hitc.com/baseline/.

3.2. Storage Device Layouts

Storage Device layouts, disk partitions, vary from DAAC to DAAC based on the amount of data storage expected to be needed to accommodate a particular DAAC's storage requirements. Disk partitions for the ACL server at each DAAC is detailed in the following documents:

TBS

These documents are maintained as part of the ECS baseline and available on the world-wide web at the URL http://pete.hitc.com/baseline/.

This page intentionally left blank.

4. Data Design

4.1. Database Overview

ACL is designed to be consistent with the Distributed Computing Environment (DCE) security paradigm. DCE is an object -oriented paradigm by design. Security is obtained by controlling a principal's (user's) access to these objects. Groups of related objects may be classified into types. A separate ACL database is created for each object type. This database contains the access control information for each object belonging to that object type. In ECS, for example, the Advertising Server provides a number of functions for which access control is desired: submitting an advertisement, approving an advertisement. Under DCE, the Advertising Server itself could be designated as an object type, each of these functions submitting an advertisement or approving an advertisement, for which access control is desired, , could be designated as an object for which there is an access control list.

The ACL database implements the large majority of the persistent data requirements for the ACL subsystem. The database is designed in such a manner as to satisfy business policy while maintaining data integrity and consistency. Database tables are implemented using the Sybase Relational Database Management system (DBMS) version 11.0.1. All components of the ACL database are described in the sections which follow in sufficient detail to support maintenance needs.

Please Note: The ACL database, while available in Drop 3.0, it is not populated and utilized until Drop 4.0

4.1.1. Physical Data Model Entity Relationship Diagram

The Entity Relationship Diagram(ERD) presents a schematic depiction of the ACL physical data model. The ERDs presented here for the ACL database were produced using the S-Designor Data Architect Computer Aided Software Engineering (CASE) tool. ERDs represent the relationship between entities or database tables. The key for the symbols used in the ERDs follows.

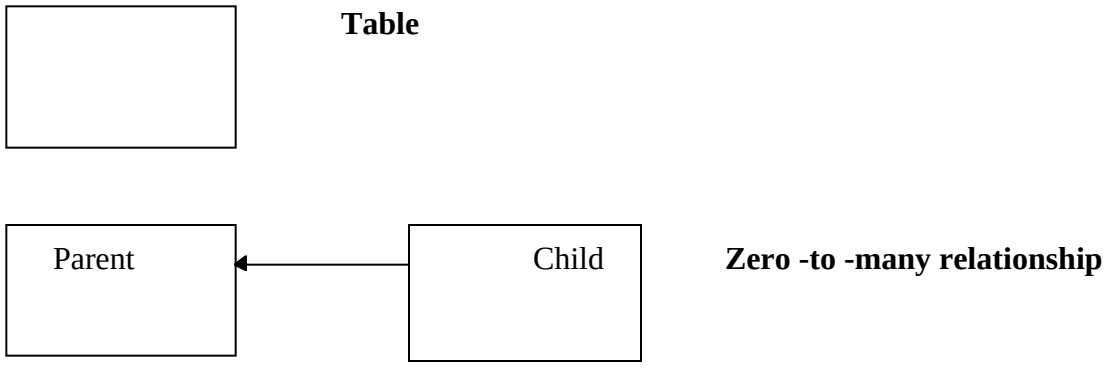

Figure 4-1. ERD Key

The ERDs for the ACL database are shown in Figures 4-2.

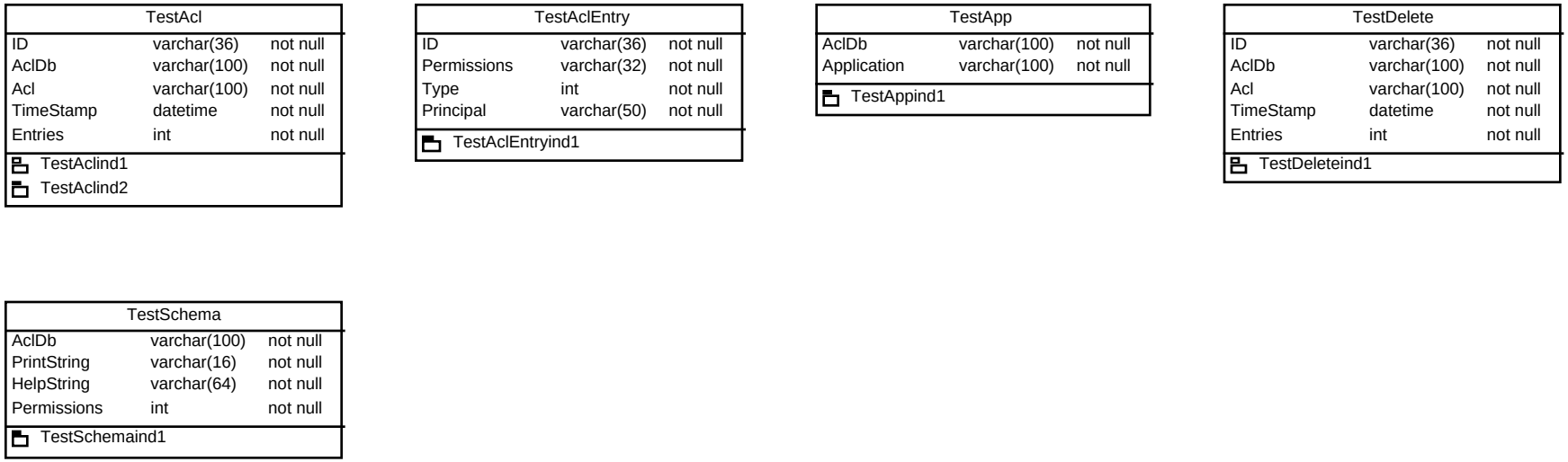

Figure 4-2. ACL ERD

4.1.2. Tables

A listing of each the tables in the ACL database is given here. A brief definition of each of these tables follows.

Table Name
TestAcl
TestAclEntry
TestApp
TestDelete
TestSchema

Table: TestAcl

Description

This table holds the Access Control List information for each object.

Table Layout:

Column	Type	Primary key	Mandatory
Acl	varchar(100)	No	Yes
AclDb	varchar(100)	No	Yes
Entries	int	No	Yes
ID	varchar(36)	No	Yes
TimeStamp	datetime	No	Yes

Table: TestAclEntry

Description

This table holds the object permissions granted to a specific principal.

Table Layout

Column	Type	Primary Key	Mandatory
ID	varchar(36)	No	Yes
Permissions	varchar(32)	No	Yes
Principal	varchar(50)	No	Yes
Type	int	No	Yes

Table: TestApp

Description

Maps an ECS application to a specific AccessControl List database (object type).

Table Layout

Column	Type	Primary Key	Mandatory
AclDb	varchar(100)	No	Yes
Application	varchar(100)	No	Yes

Table: TestDelete

Description

This table holds all the deleted Access Control List entries.

Table Layout

Column	Type	Primary key	Mandatory
Acl	varchar(100)	No	Yes
AclDb	varchar(100)	No	Yes
Entries	int	No	Yes
ID	varchar(36)	No	Yes
TimeStamp	datetime	No	Yes

Table: TestSchema

Description

This table holds a definition of the permissions that are valid for a givento object type.

Table Layout

Column	Type	Primary Key	Mandatory
AclDb	varchar(100)	No	Yes
HelpString	varchar(64)	No	Yes
Permissions	int	No	Yes
PrintString	varchar(16)	No	Yes

4.1.3. Columns

Brief definitions of each of the columns present in the database tables defined above are contained herein.

Column: Acl

Description

This holds the name of the object for which access control is required.

Constraints:

Column: AclDb

Description

The object type or classification to which the object belongs.

Constraints:

Column: Application

Description

The name of an ECS application for which access control is required.

Constraints:

Column: Entries

Description

The number of times that a specific object is being connected to.

Constraints:

Column: HelpString

Description

Holds the meaning of each permission type for a given object type

Constraints:

Column: ID

Description

This the ACL identifier.

Constraints:

Column: Permissions (TestAclEntry table)

Description

It holds the permission type applicable to a specific object type.

Constraints:

Column: Principal

Description

The name or identifier of a principal requiring access to objects.

Constraints:

Column: PrintString

Description

The valid permission options to be displayed for a given object type.

Constraints:

Column: TimeStamp

Description

The time that a ACL entry was inserted or modified.

Constraints:

Column: Type

Description

The classification code given to each type of principal.

Constraints:

4.1.4. Column Domains

Domains specify the ranges of values allowed for a given table column. Sybase supports the definition of specific domains to further limit the format of data for a given column. Sybase domains are, in effect, user-defined data types. There are no domains defined in the ACL database.

4.1.5. Rules

Sybase supports the definitions of rules. Rules provide a means for enforcing domain or validation criteria constraints on a given column. There are no rules defined in Sybase for the ACL database.

4.1.6. Defaults

Defaults are used to supply a value for a column when one is not defined at insert time. There are no defaults defined in Sybase for the ACL database.

4.1.7. Views

Sybase allows the definition of views as a means of limiting an application or users access to data in a table or tables. Views create a logical table from columns found in one or more tables. There are no views defined in Sybase for the ACL database.

4.1.8. Integrity Constraints

Sybase version 11.0.1 allows the enforcement of referential integrity via the use of declarative integrity constraints. Integrity constraints allow the SQL server to enforce primary and foreign key integrity checks without automatically without requiring programming. Sybase 11 is only ANSI-92 compliant, however, therefore its constraints support "restrict-only" operations. This means that a row can not be deleted or updated if their are rows in other tables having a foreign key dependency on that row. Cascade delete and update operations can not be performed if a declarative constraint has been used. There are no declarative integrity constraints defined in the ACL database.

4.1.9. Triggers

Sybase supports the enforcement of business policy via the use of triggers. A trigger is best defined as set of activities or checks that should be performed automatically when ever a row is inserted, updated, or deleted from a given table. Sybase version 11.0.1 allows the definition of insert, update, and delete trigger per table. There are no triggers currently defined in the ACL database..

4.1.10. Stored Procedures

Sybase also includes support for business policy via the use of stored procedures. Stored procedures are typically used to capture a set of activities or checks that will be performed on the database repeatedly to enforce business policy and maintain data integrity. Stored procedures are parsed and complied SQL code that reside in the database and may be called by name by an application, trigger or another stored procedure A listing of each the stored procedures and a brief definition of each follows.

Procedure List

Name	Description
EcSeGetAclInfoProc	Retrieves select Access Control List data for a given application.
EcSeGetAllAclInfoProc	Retrieves all Access Control List data for a given application
EcSeGetSchemaInfoProc	Retrieves permission (access) types valid for a given application.

Procedure: EcSeGetAclInfoProc

Code

```
CREATE PROC EcSeGetAclInfoProc
        (@Applicationvarchar(100) = NULL,   /* Application name to look up */
```

```
        @TimeStamp datetime = NULL,      /* Required if flag is set to 1 */
        @Flag           int = 0)/* Set to 1 or 0; 1 - check TimeStamp */
AS
BEGIN
  /* @Application is required to limit the Acl(s) selected. */
  IF @Application = NULL
    BEGIN
        /* mimic Sybase Error */
        RAISERROR 20010 "Procedure EcSeGetAclInfoProc expects parameter
@Application, which was not supplied."
        RETURN -999
    END
  ELSE
    BEGIN
        /* If @Flag = 1, then @TimeStamp must be provided. */
      IF @Flag = 1 AND @TimeStamp = NULL
        BEGIN
            /* mimic Sybase Error */
            RAISERROR 20011 "Procedure EcSeGetAclInfoProc expects parameter
@TimeStamp, which was not supplied."
            RETURN -999
        END
      ELSE
          BEGIN
         IF @Flag = 1
              BEGIN
                SELECT      b.AclDb, b.Acl, c.Principal,
                            rtrim(convert(varchar(10), c.Type)),
                            c.Permissions
                FROM        TestApp a, TestAcl b, TestAclEntry c
                WHERE       (a.AclDb = b.AclDb              AND
                             a.Application = @Application)        AND
                            b.ID = c.ID                     AND
                            b.TimeStamp >= @TimeStamp
                ORDER BY b.AclDb, b.Acl
              END
            ELSE        /* If @Flag is not = 1, do not compare the TimeStamp. */
              BEGIN
                SELECT      b.AclDb, b.Acl, c.Principal,
                            rtrim(convert(varchar(10), c.Type)),
                            c.Permissions
                FROM        TestApp a, TestAcl b, TestAclEntry c
                WHERE       (a.AclDb = b.AclDb              AND
                             a.Application = @Application)  AND
                            b.ID = c.ID
                ORDER BY b.AclDb, b.Acl
              END
          END
    END
END
go
```

Procedure: EcSeGetAllAclInfoProc

Code

```
CREATE PROC EcSeGetAllAclInfoProc
        (@Applicationvarchar(100) = NULL,   /* Application name to look up */
         @TimeStamp datetime = NULL,      /* Required if flag is set to 1 */
         @Flag          int = 0)       /* Set to 1 or 0; 1 - check TimeStamp */
AS
BEGIN
  /* @Application is required to limit the Acl(s) selected. */
  IF @Application = NULL
    BEGIN
        /* mimic Sybase Error */
        RAISERROR 20015 "Procedure EcSeGetAllAclInfoProc expects parameter
@Application, which was not supplied."
        RETURN -999
    END
  ELSE
    BEGIN
        /* If @Flag = 1, then @TimeStamp must be provided. */
     IF @Flag = 1 AND @TimeStamp = NULL
       BEGIN
           /* mimic Sybase Error */
           RAISERROR 20016 "Procedure EcSeGetAllAclInfoProc expects parameter
@TimeStamp, which was not supplied."
           RETURN -999
       END
     ELSE
         BEGIN
        SELECT      a.AclDb, a.Application,
                    b.PrintString, b.HelpString, b.Permissions
        INTO        #TempTblOne
   FROM       TestApp a, TestSchema b
        WHERE       a.AclDb = b.AclDb AND
                    a.Application = @Application

        IF @Flag = 1
            BEGIN
              SELECT        a.ID, a.AclDb, a.Acl, a.TimeStamp, a.Entries,
                            b.Permissions, b.Type, b.Principal
              INTO          #TempTblTwo
              FROM  TestAcl a, TestAclEntry b, TestApp c
              WHERE         a.ID = b.ID AND
                            (c.Application = @Application AND
                             c.AclDb = a.AclDb) AND
                             a.TimeStamp >= @TimeStamp
            END
          ELSE        /* If @Flag is not = 1, do not compare the TimeStamp. */
            BEGIN
              SELECT        a.ID, a.AclDb, a.Acl, a.TimeStamp, a.Entries,
                            b.Permissions, b.Type, b.Principal
              INTO          #TempTblThree
```

```
            FROM  TestAcl a, TestAclEntry b, TestApp c
            WHERE       a.ID = b.ID AND
                        (c.Application = @Application AND
                         c.AclDb = a.AclDb)
          END

        IF @Flag = 1
            BEGIN
              SELECT    a.AclDb, a.Application, a.PrintString,
                        a.HelpString, a.Permissions,
                        b.ID, b.Acl, b.TimeStamp, b.Entries,
                        b.Permissions, b.Type, b.Principal
              FROM              #TempTblOne a, #TempTblTwo b
              WHERE     a.AclDb = b.AclDb
            END
        ELSE
            BEGIN
            SELECT      a.AclDb, a.Application, a.PrintString,
                        a.HelpString, a.Permissions,
                        b.ID, b.Acl, b.TimeStamp, b.Entries,
                        b.Permissions, b.Type, b.Principal
              FROM              #TempTblOne a, #TempTblThree b
              WHERE     a.AclDb = b.AclDb
            END
      END
  END
END
go
```

Procedure: EcSeGetSchemaInfoProc

Code

```
CREATE PROC EcSeGetSchemaInfoProc
        (@Applicationvarchar(100) = NULL)   /* Application name to look up */
AS
BEGIN
  /* @Application is required to limit the Acl(s) selected. */
  IF @Application = NULL
    BEGIN
        /* mimic Sybase Error */
        RAISERROR 20020 "Procedure EcSeGetSchemaInfoProc expects parameter
@Application, which was not supplied."
        RETURN -999
    END
  ELSE
    BEGIN
        SELECT              a.AclDb,
                    b.PrintString,
                    b.HelpString
        FROM                TestApp a,
                    TestSchema b
        WHERE               (a.AclDb = b.AclDb          AND
                     a.Application = @Application)
        ORDER BY   a.AclDb
    END
END
go
```

4.2. File Usage

There are cases when the implementation of a persistent data requirement is better suited to a flat file than to a database table. A typical example of such data is system configuration information. System configuration information is fairly static and usually has no explicit relationship to other data in the enterprise. Another common use of files in ECS is as an interface mechanism between ECS and the external world. ACL file usage is detailed in this section via file definitions, attribute definitions, and attribute domain definitions.

4.2.1. Files Definitions

A listing of each the files in the ACL database is given here. A brief definition of each of these files follows.

TBS.

4.2.2. Attributes

Brief definitions of each of the attributes present in the files defined above are contained herein.

TBS.

4.2.3. Attribute Domains

Domains represent the ranges of valid values allowed for a given file attribute. Attributes domains for each of the attributes defined above are given here.

TBS

5. Performance and Tuning Factors

5.1. Indexes

An index provides a means of locating a row in a table based on the value of specific a columns, without having to scan each row in the table. If used appropriately, indexes can significantly increase data retrieval. Sybase allows the definition of two types of indexes, clustered and non-clustered. In a clustered index, the rows in a tables are physically stored in the sort order determined by the index. Clustered indexes are particularly useful, when the data is frequently retrieved in order. Non-clustered indexes differ from their clustered counterpart, in that data is not physically stored in sort order. Only one clustered index may be defined per table. All of the indexes defined against tables in the ACL database are described herein.

TBS

5.2. Segments

Sybase supports the definition of segments. A segment is a named pointer to a storage device or devices. Segments are used to manually place database objects onto particular storage devices. All segments explicitly defined in the ACL database are described herein.

TBS

5.3. Named Caches

A cache is a block of memory that is used by Sybase to house data pages that are currently being accessed. A named cache is a named block of memory that the SQL server can use to house frequently accessed tables. Assigning a table to cache causes it to be loaded into memory. This greatly increases performance by eliminating the time expense normally associated with disk i/o. Named caches used in the ACL databases are described herein.

TBS

This page intentionally left blank.

6. Database Security

6.1. Initial Users

Upon initial installation the following users will have access to ACL database. The level of access is limited to that associated with their assigned group and/or role. A complete definition of each if these groups and roles is maintained in the following documents:

920-TDG-009 Release B0 GSFC DAAC Database Information

920-TDN- 009 Release B0 NSIDC DAAC Database Information

920-TDE-009 Release B0 EDC DAAC Database Information

920-TDL-009 Release B0 LARC DAAC Database Information

920-TDS-009 Release B0 SMC DAAC Database Information

920-TDM-009 Release B0 Mini-DAAC Database Information

6.2. Groups

Groups are a means of logically associating users with similar data access needs. Once a group has been defined, object and command permissions can be granted to that group. A user who is member of a group inherits all of the permissions granted to that group. Several group have been defined in the ACL database upon initial installation. A definition of each of these groups is contained herein.

TBS

6.3. Roles

Roles were introduced in Sybase 10 to allow a structured means for granting users the permissions needed to perform standard database administration activities and also provide a means for easily identifying such users. There a six pre-defined roles that may be assigned to a user. A definition of each of these roles follows as well as a description of the types of activities that may be performed by each role.

TBS

System Administrator (sa_role) - This role is used to grant a specific user to permissions needed to perform standard system administrator duties including:

- installing SQL server and specific SQL server modules
- managing the allocation of physical storage
- tuning configuration parameters
- creating databases

Site Security Officer (sso_role) - This role is used to grant a specific user the permissions needed to maintain SQL server security including:

- adding server logins
- administrating passwords
- managing the audit system
- granting users all roles except sa_role

Operator (oper_role) - This role is used to grant a specific user the permissions needed to manage backup and recovery of the database including;

- dumping transactions and databases
- loading transactions and databases

Navigator (navigator_role) -This role is used to grant a specific user the permissions needed to manage the navigation server.

Replication (replication_role) - - This role is used to grant a specific user the permissions needed to manage the replication server.

Sybase Technical Support (sybase_ts_role) - This role is used to grant a specific user the permissions needed to perform database consistency checker (dbcc), a sybase supplied utility, commands that are considered outside of the realm of normal system administrator activities.

6.4. Object Permissions

TBS

7. Replication

7.1. Replication Overview

Replication as the name implies is a set of Sybase products that allow replication of data from one database to another. The ACL database employs replication to support its REPTYPE requirements. In order for replication to be accomplished the data source must defined the tables and columns that may be replicated to a data recipient. These definitions are referred to replication definitions. In the same manner a data recipient must specify the replication definitions in which he is interested. These specifications are referred to as replication subscriptions. In addition the replication database and server must be configured to support the potentially large volumes of data that will be transferred between the source and recipient databases. Replication is not currently slated for ACL.

7.2. Replication Definitions

Not applicable.

7.3. Replication Subscriptions

Not Applicable.

7.4. Replication Database Configuration

Not Applicable.

7.5. Replication Server Configuration

Not Applicable.

This page intentionally left blank.

8. Scripts

8.1. Installation Scripts

Any scripts used to support installation of the ACL database are described herein.

TBS

8.2. De-Installation Scripts

Any scripts used to support de-installation of the ACL database are described herein.

TBS

8.3. Backup/Recovery Scripts

Any scripts used to facilitate backup or recovery of the ACL database are described herein.

TBS

8.4. Miscellaneous Scripts

Miscellaneous scripts applicable to the ACL database are described herein.

TBS

This page intentionally left blank.

Abbreviations and Acronyms

ACL	Access Control List
ACMHW	Access and Control Management HWCI
ADC	affiliated data center
ADSHW	Advertising Server HWCI
ADSRV	Advertising Service CSCI
AI&T	algorithm integration and test
AITHW	Algorithm Integration and Test HWCI
AITTL	Algorithm Integration and Test CSCI
AM-1	EOS AM Project spacecraft 1, morning spacecraft series -- ASTER, CERES, MISR, MODIS and MOPITT instruments
ANSI	American National Standards Institute
API	application program (or programming) interface
APID	application's process ID
AQAHW	Algorithm QA HWCI
ASCII	American Standard Code for Information Exchange
ASTER	Advanced Spaceborne Thermal Emission and Reflection Radiometer (formerly ITIR)
AVHRR	Advanced Very High-Resolution Radiometer
BER	bit error rate
BUFR	binary universal format for representation of data
CASE	Computer Aided Software Engineering
CCSDS	Consultative Committee for Space Data Systems
CD	contractual delivery 214-001
CD-ROM	compact disk -- read only memory
CDR	Critical Design Review
CDRL	contract data requirements list
CERES	Clouds and Earth's Radiant Energy System

CI	configuration item
COTS	commercial off-the-shelf (hardware or software)
CPU	central processing unit
CSCI	computer software configuration item
CSDT	Computer Science Data Type
CSMS	Communications and Systems Management Segment (ECS)
CSS	Communications Subsystem
DAAC	Distributed Active Archive Center
DAN	data availability notice
DAO	Data Assimilation Office
DAR	data acquisition request
DAS	data availability schedule
DBMS	Database Management System
DDICT	Data Dictionary CSCI
DDIST	Data Distribution Services CSCI
DDSRV	Document Data Server CSCI
DESKT	Desktop CSCI
DID	data item description
DIM	distributed information manager (SDPS)
DIMGR	Distributed Information Manager CSCI
DIPHW	Distribution and Ingest Peripheral Management HWCI
DMGHW	Data Management HWCI
DMS	Data Management Subsystem
DMWG	Data Management Working Group
DP	Data Provider
DPR	data processing request
DPREP	Science Data Preprocessing CSCI
DPS	Data Processing Subsystem
DRPHW	Data Repository HWCI
DSS	Data Server Subsystem

ECS	EOSDIS Core System
EDC	EROS Data Center
EDHS	ECS Data Handling System
EDOS	EOS Data and Operations System
EOS	Earth Observing System
EOS-AM	EOS Morning Crossing (Descending) Mission -- see AM-1
EOSDIS	Earth Observing System Data and Information System
EROS	Earth Resources Observation System
ESDIS	Earth Science Data and Information System (GSFC)
ESDT	Earth science data types
ESN	EOSDIS Science Network (ECS)
FDDI	fiber distributed data interface
FDF	flight dynamics facility
FDFEPHEM	FDF-generated definitive orbit data
FGDC	Federal Geographic Data Commuittee
FK	Foreign Key
FOO	Flight of Opportunity
FOS	Flight Operations Segment (ECS)
GB	gigabyte (10^9)
GNU	(recursive acronym: “GNU’s Not Unix”); a project supported by the Free Software Foundation dedicated to the delivery of free software
GPCP	Global Precipitation Climatology Project
GPCP	Global Precipitation Climatology Project
GPI	GOES Precipitation Index
GRIB	GRid In Binary
GSFC	Goddard Space Flight Center
GTWAY	Version 0 Interoperability Gateway CSCI
GUI	graphic user interface
GV	ground validation
HDF	hierarchical data format

HDF-EOS	an EOS proposed standard for a specialized HDF data format
HIPPI	high performance parallel interface
HMI	human machine interface
HTML	HyperText Markup Language
HTTP	Hypertext Transport Protocol
HWCI	hardware configuration item
I&T	integration and test
I/F	interface
I/O	input/output
ICD	interface control document
ICLHW	Ingest Client HWCI
ID	identification
IDE	Interactive Development Environments
IDG	Infrastructure Development Group
IDR	Incremental Design Review
IERS	International Earth Rotation Service
IMS	Information Management System (obsolete ECS element name)
INGST	Ingest Services CSCI
IOS	Interoperability Subsystem
IP	international partners
IR-1	Interim Release 1
IRD	interface requirements document
ISO	International Standards Organization
ISS	Internetworking Subsystem
IV&V	independent verification and validation
JPL	Jet Propulsion Laboratory
L0-L4	Level 0 (zero) through Level 4
LaRC	Langley Research Center (DAAC)
LIM	local information manager (SDPS)

LIMGR	Local Information Manager CSCI
LIS	Lightning Imaging Sensor
LSM	local system management (ECS)
MB	megabyte (10^6)
MDT	mean downtime
MDT	mean downtime
MFLOPS	mega (millions of) floating-point operations (10^6) per second
MISR	Multi-Angle Imaging SpectroRadiometer
MODIS	Moderate-Resolution Imaging Spectrometer
MOPITT	Measurements of Pollution in the Troposphere
MSFC	Marshall Space Flight Center
MSS	Management Support Subsystem
MTBF	mean time between failure
MTPE	Mission to Planet Earth
MTTR	mean time to restore
N/A	not applicable
NAS	National Academy of Science
NASA	National Aeronautics and Space Administration
NESDIS	National Environmental Satellite Data and Information Service
NMC	National Meteorological Center (NOAA)
NOAA	National Oceanic and Atmospheric Administration
NSIDC	National Snow and Ice Data Center (DAAC)
O/A	orbit/altitude
ODC	other data center
OSI	Open System Interconnect
PDPS	Planning and Data Processing Subsystem
PDR	Preliminary Design Review
PDS	production data set
PGE	Product Generation Executive

PGS	Product Generation System (obsolete ECS element name) (ASTER)
PK	Primary Key
PLANG	Production Planning CSCI
PLNHW	Planning HWCI
PLS	Planning Subsystem
POSIX	Portable Operating System Interface for Computer Environments
PR	Precipitation Radar (TRMM)
PRONG	Processing CSCI
QA	quality assurance
RMA	reliability, maintainability, availability
RTF	rich text format
SAA	satellite active archive
SAGE	Stratospheric Aerosol and Gas Experiment
SCF	Science Computing Facility
SDP	Science Data Processing
SDPF	Sensor Data Processing Facility (GSFC)
SDPS	Science Data Processing Segment (ECS)
SDPTK	SDP Toolkit CSCI
SDSRV	Science Data Server CSCI
SeaWIFS II	Sea-Viewing Wide Field-of-View Sensor II
SFDU	Standard Format Data Unit
SMC	System Management Center (ECS)
SPRHW	Science Processing HWCI
SRS	software requirements specification
SSM/I	Special Sensor for Microwave/Imaging (DMSP)
SST	sea surface temperature
STMGMT	Storage Management
STMGT	Storage Management Software CSCI
SUBSRV	Subscription Server

TMI	TRMM Microwave Image
TOMS	Total Ozone Mapping Spectrometer
TONS	TDRS On-board Navigational System
TRMM	Tropical Rainfall Measuring Mission (joint US-Japan)
TSDIS	TRMM Science Data and Information System
USNO	US Naval Observatory
UT	universal time
UTC	universal time code
V0	Version 0
VIRS	Visible Infrared Scanner (TRMM)
WAIS	Wide Area Information Server
WKBCH	Workbench CSCI
WKSHW	Working Storage HWCI
WWW	World-Wide Web

This page intentionally left blank.